

NooJ's dictionaries

Max Silberztein

Université de Franche-Comté

Abstract

NooJ is the new linguistic development environment developed by Max Silberztein. It has been designed to replace INTEX in the near future. NooJ has been rewritten from the ground up, with a new computer architecture and an innovative linguistic engine that makes it possible to build a new set of NLP applications. Among other novelties, NooJ's linguistic engine uses a new type of lexical data base, its dictionaries, that are an improvement upon the INTEX dual system of DELA dictionaries and Lexicon-Grammar tables.

Introduction

The INTEX linguistic engine (Silberztein 1993) used two sets of dictionaries that were designed at Prof. Maurice Gross's LADL laboratory: on the one hand, DELA-type dictionaries describe simple and compound words (Courtois, Silberztein 1990); on the other hand, lexicon-grammars describe frozen expressions (Gross M. 1993).

The INTEX lexical parser recognizes simple words by looking up DELAF-type dictionaries, whose entries are all the inflected word forms of a language (Courtois 1990). For instance, the following is a sample of a DELAF:

```
had , have . V : I
has , have . V : P3s
have , have . V : W
having , have . V : G
```

Each entry (e.g. "had") is associated with its lemma ("have"), a category ("V") and a series of inflectional codes ("I" stands for "Preterit"). DELAFs are themselves constructed automatically from DELAS-type dictionaries, whose entries are lemmas associated with an inflectional paradigm that allows INTEX to automatically generate the DELAFs.

In the same manner, INTEX recognizes compound words by a lookup of DELACF-type dictionaries, in which inflected forms of each compound word are explicitly listed (Silberztein 1990). For instance following is a sample of a DELACF dictionary:

```
abandoned ship , abandoned ship . N : s
abandoned ships , abandoned ship . N : p
act of God , act of God . N : s
acts of God , act of God . N : p
gingerbread men , gingerbread man . N : p
```

In INTEX, DELACF-type dictionaries are constructed semi-automatically from DELACs (similarly to the DELAS/DELAF system), through a

complex series of INTEX, PERL and manual editing operations.

Recognizing frozen expressions is more complex. Frozen expressions are described in Lexicon-grammars (Gross M. 1993). Lexicon-Grammars are displayed as tables, in which each line corresponds to a lexical entry, and each column corresponds to a property. Property values are either binary ("+" or "-"), or textual (e.g. **PREP** = "for").

In order to automatically recognize frozen expressions in texts, INTEX uses slightly adapted lexicon-grammar tables, in conjunction with **meta-graphs** (Silberztein 1999) (Vietri 2004). For instance, the following is a sample of a lexicon-grammar table that represents French frozen expressions of the **N0 V N1** syntactic structure (e.g. *John kicked the bucket*):

	A	B	C	D	E	F	G	H	I	J	K
1	N0 = Noun	N0 = Noun	Neg	PPV	V	NO V	DET	NO V Det N1	N	N1 = Npe	Paraff
72	+	+	-	<E>	<attendre>	-	le	-	résultat de l'enquête	-	+
73	+	+	-	<E>	<attirer>	-	l'	-	oeil	-	+
74	+	+	-	<E>	<attraper>	-	le	-	ballon	-	+
75	+	+	-	<E>	<attraper>	-	le	-	coup	-	+
76	+	+	-	<E>	<avalier>	-	la	-	fumée	-	+

Figure 1: A lexicon-grammar table for INTEX

INTEX meta-graphs are context-free grammars (CFGs) that contain links to cells in the table. For instance, in the meta-graph associated with the previous figure, the symbol @E corresponds to the 5th column "V"; thus its value is the verb of each frozen expression. Paths in the CFG grammar may be enabled or disabled in accordance with the value of binary property cells in the lexicon-grammar table ("+" or "-"). For instance, the symbol @F in the meta-graph associated with the previous figure corresponds to the 6th column "NO V"; its value,

which is either a “+” or a “-”, enables or disables the path in the CFG that recognizes the intransitive structure for each expression.

Problems

Problems with the INTEX approach have been discussed extensively in (Silberstein 2004). The most important are:

-- the separation of DELAS-F and DELAC-F dictionaries is artificial, and has a number of unpleasant consequences. For instance, we need to process “priori” (from the adverb “a priori”) as a lexical entry of a DELAS; there are hundreds of such “words”. A large number of terms have parts of their orthographical variants stored in DELAS dictionaries, and others in DELACs; e.g. “pianoforte” is stored in a DELAS, whereas “piano forte” is stored in a DELAC. INTEX is not capable of processing compounds inflection automatically, etc.

-- the pair (lexicon-grammar table, meta graph) is awkward to use. Meta-graphs cannot be re-used with other INTEX syntactic grammars. They become more and more complex as semi-frozen expressions accept aspectual variants and different sets of syntactic structures, to a point where they would be totally impractical to use in conjunction with the lexicon-grammar tables of free verbs ⁽¹⁾.

-- DELAS/DELAC dictionaries are not adapted to either multilingual applications or semantic applications in which lexical entries are associated with synonymous terms or expressions, hyperonyms, etc (Koeva S., Mihov S. 2005). Not to mention that INTEX needs five different types of dictionaries, each one with its own formalism...

NooJ’s dictionaries

In NooJ, these five dictionaries are represented in one unique format. In order to unify the formalism, we needed to (1) merge simple and compound words (2) get rid of DELAFs and DELACFs and (3) add information to DELA-type dictionaries so that they can be processed in the same way as lexicon-grammar tables.

1. Merging simple and compound words

NooJ dictionaries contain indistinctly simple or compound words thanks to a new inflection system that can process both simple and compound words’ inflectional morphology in a unified way.

¹ (Roche 1993) does describe the formalization of one lexicon-grammar table with meta-graphs that are similar to INTEX’s, but the size of the resulting grammar and the number of grammars that would have to be implemented (there are a hundred tables) makes this approach extremely difficult to implement.

For instance, the two following lexical entries:

```
academic program,N+FLEX=APPLE  
window,N+FLEX=APPLE
```

inflect the same way (they take an ‘s’ in the plural). Therefore, they are both associated with the same inflectional class: APPLE. The class APPLE is defined by the following expression:

```
APPLE = <E>/singular + s/plural;
```

that states that if one adds nothing to the lexical entry (“<E>” is the empty string), one gets the singular form (“singular”); if one adds an “s” to the end of the lexical entry, one gets the plural form (“plural”).

NooJ’s inflectional engine is equivalent to a stack automaton. It uses a dozen default commands that operate on the suffix of lexical entries, as follows:

```
<B>: equivalent to the keyboard key “Backspace”  
<D>: Duplicate current character  
<E>: Empty string  
<L>: equivalent to the keyboard key “Left arrow”  
<N>: go to end of Next word form  
<P>: go to end of Previous word form  
<R>: equivalent to the keyboard key “Right arrow”  
<S>: equivalent to the keyboard “delete” key
```

Users can override these commands, and add their own: for instance, we have added operators such as <A> (“remove accent”) and <À> (“add a grave accent”) to the inflectional engine of Romance languages, and the operator <F> (“Finalize letter”) for Hebrew. ⁽²⁾

In contrast with INTEX, NooJ is capable of inflecting compounds. For instance, the class “ACTOFGOD” is defined by the following expression:

```
ACTOFGOD = <E>/singular + <PW>s/plural;
```

The operator <PW> stands for: “go to the end of the first component of the lexical entry”. Note that the following three entries are associated with this class, even though their length is different:

```
bag of tricks,N+FLX=ACTOFGOD  
balance of payment deficit,N+FLX=ACTOFGOD  
member of the opposite sex,N+FLX=ACTOFGOD
```

² NooJ contains a dozen labs aimed at demonstrating its functionalities. Check out its inflectional lab.

In the same manner, even though the lexical entries *blank piece of paper*, *last line of defence*, *family history of cancer*, *sexual harassment in the work place*, etc. have different lengths, they can be associated with a unique inflectional class because the inflection is carried by the same component (the second one).

Agreements between components of a compound can be described as well. For instance, the following inflectional expression formalizes the agreement between the two components of compounds such as *journeyman carpenter*:

```
<E>/singular + s<P><B2>en/plural
```

To get the plural form, add an “s” to the end of the compound, then go back to the previous (<P>) component, delete the two last characters (<B2>), and add the suffix “en”.

In conclusion, NooJ can process simple and compound words’ inflection completely automatically, whereas INTEX could only process simple words automatically. This has allowed us to unify the description of simple and compound words.

2. No more DELAFs

NooJ’s lexical parser requires no DELAF/DELACF-type dictionaries. ⁽³⁾

NooJ dictionaries, which are in effect similar to DELAS (or DELAC) dictionaries, are compiled into a Finite-State Transducer that is more powerful than previous DELAF transducers, and faster in some cases.

This characteristic is more important than a mere increase in comfort for the users, or even an increase in efficiency: encoding the inflectional information inside NooJ dictionaries gives NooJ the ability to perform morphological operations **during parse time**. Whereas INTEX could link an inflected form to its lemma (by a lookup of the DELAF transducer), NooJ can link any inflected form to any other inflected form. Thus, NooJ can perform complex transformations within texts. For instance, it is now possible to replace a certain conjugated verb with its past participle form, and vice-versa, within a particular text:

³ For compatibility reasons, users can still import their own DELAF-DELACF into NooJ. However, if the inflectional system is not integrated in NooJ, NooJ will not be able to perform morphological operations during parse time.

John eats the apple <=> *the apple is eaten by John*

Using this new functionality will enhance several current NLP applications, such as automatic translation and information retrieval applications.

3. Unifying Lexicon-Grammar tables and DELA dictionaries

On the one hand, lexicon-grammar tables are associated with a predefined set of properties: all the entries of a table must be described in terms of the same exact properties. Any modification or addition of a property requires modification of the whole table and associated meta-graph, as well as the recompilation of the resulting CFG grammar.

On the other hand, DELA-type dictionaries have open sets of properties: in one given DELA dictionary properties may change from one lexical entry to another. They can be edited, and new properties specific to an application may be added at any moment without having to recompile the dictionary, because all INTEX functionalities (indexing, concordances, disambiguation, syntactic parsing, etc.) take lexical properties into account **at run-time**.

Naturally, the openness of DELA-type dictionaries also has disadvantages: properties cannot be checked, and it is difficult for lexicographers to maintain more than a few properties.

NooJ dictionaries combine the advantages of both lexicon-grammar tables and DELA-type dictionaries: they can be displayed either in list form or in table form. This has led us to adapt both the notation of lexicon-grammar tables and DELA-type dictionaries.

Importing a lexicon-grammar table in NooJ is straightforward ⁽⁴⁾, but it requires rendering all the properties that are common for all the entries of the table explicit. Each table of the lexicon-grammar is indeed defined by a set of syntactic properties, see (Leclère 2002). For instance, the table “4” which is described in (Gross 1975), is associated with verbs of the transitive structure **N0 V N1**, in which **N0** (the subject) is an abstract noun or a sentence, and **N1** (the object) is a Human noun, e.g.:

⁴ lexicon-grammar tables need to be slightly adapted to be used by INTEX or NooJ. For instance, the “-” character (binary “false”) had to be distinguished from the “<E>” symbol (empty string), words that could be inflected (between angles) had to be distinguished from constant word forms, etc.

This problem upsets John
That it rained all day amuses Mary

Therefore, in NooJ all the entries of table 4 are associated with the explicit properties: **NVN** (to state that the verb enters in the transitive syntactic structure), **N0=Abst** and **N1=Hum**.

When a lexicon-grammar table has been adapted to NooJ, displaying it in a list view is straightforward: each row of a table corresponds to a line in the list; non-negative cells are prefixed with a character “+”, e.g. “+NVN”; textual cells are transcribed into a pair Property=Value, e.g. “+PREP=**from**”; negative cells are simply ignored.

Importing a traditional DELA dictionary is more complex: it requires that features of the dictionary (that correspond to property values of a lexicon-grammar) be typed, so that all the values of a common property can be regrouped in one column. We also need to state the number of relevant properties for each category of word (e.g. **Tense** for Verbs, **Number** for Nouns, etc.), in order to distinguish absent default values, from irrelevant ones.

This is done via a “Property Definition” file that contains rules such as:

```
N_Distribution = Hum + Conc + Abst ;
N_Gender = m + f ;
N_Number = s + p ;
...
V_Tense = Present + Futur +... ;
V_Pers = 1 + 2 + 3 ;
V_Number = s + p ;
...
```

Note that all features in a NooJ dictionary do not have to be typed; if NooJ does not know the type of a feature, it will simply display it as a column header, and enter the “+” (if the feature is present) and “-” (if absent) values accordingly.

A given property may be associated with more than one category (e.g. **Number** is relevant both for nouns and verbs). But NooJ checks that one feature (e.g. “+p”) does not correspond to more than one property (e.g. “**Gender**” and “**Tense**”).

The next figure displays a NooJ dictionary in table form:

Entry	Category	CR	Genre	Nombre	Pers	Sem	Temps
a	N		m	p	-		
a	N		m	s	-		
avoir	V		s		3		P
à	PREP						
abaisser	V						W
abandon	N		m	s	-		
abandonner	V		s		3		P
abandonner	V		s		2		Y
abandonner	V		s		1		P
abandonner	V		s		1		S
abandonner	V		s		3		S
abandonner	V		m	s	-		K
abandonné	A		m	s	-		
abandonné	N		m	s	-		
abandonner	V		f	s	-		K

Table view for a NooJ dictionary

NooJ distinguishes *default* properties from *irrelevant* ones: notice that in the previous figure, the **Pers** property is displayed as “-” (default) for infinitive verbs, whereas it is set as blank (irrelevant) for prepositions.

Moreover, associating NooJ lexical features with typed properties opens up possibilities for implementing unification mechanisms in the future.

By breaking the INTEX association of lexicon-grammar tables and their specific meta-graphs, we have been able to process any type of lexicon-grammar table, even the ones that describe free constructs. Meta-graphs, which have in effect been replaced with “regular” NooJ grammars, can now be reused and accumulated.

Multi-fields dictionaries

The number of fields of NooJ dictionaries is no longer limited to one! NooJ dictionaries can contain entries associated with a “super-lemma”, that can be an orthographical variant, the translation in another language, a synonymous entry or an hyperonym. For instance, consider the following lexical entries:

U.N.,United Nations,N+Org
 czar,tsar,N+FLX=Pen

The first entry (U.N.) is associated with super-lemma “United Nations”; it does not inflect. This entry is similar to a INTEX/DELACF entry.

The second entry (czar) is associated with super-lemma “tsar”; it inflects according to the paradigm “Pen” (i.e. takes an ‘s’ in the plural).

Being able to associate words with super-lemmas, i.e. words that do not necessarily correspond to their

inflectional lemma (“czar” is czars’s lemma, not “tsar”) opens up a new range of applications for NooJ.

Conclusion & Perspectives

NooJ’s system of dictionaries is a significant enhancement of INTEX’s. On the one hand, unifying DELAS, DELAF, DELAC and DELACF is a significant simplification of the former INTEX system, and results in a more powerful system thanks to its embedded morphological capabilities, as well as its possibility to associate entries with “super-lemmas”. On the other hand, the ability to manage both DELA-type dictionaries and lexicon-grammar tables in a unified way ends the artificial dichotomy that has existed for years, between two types of activities: users can now display their dictionaries as they wish, in table or list form, and can access both morphological and syntactic properties at the same time. We think that the new NooJ framework will allow us to quickly integrate the two levels of linguistic description and reach a new level of precision in NLP applications.

With the help of several NooJ users, we have already started to build a new set of dictionaries for NooJ.

See the WEB site: www.nooj4nlp.net for more information NooJ and its lexical resources.

References

- Courtois B., Silberztein, M. Eds (1990). *Dictionnaires électroniques du français*, Langue française n° 87 (127 pages). Larousse: Paris.
- Courtois, B. (1990). Le dictionnaire DELAS. In *Dictionnaires électroniques du français*, Langue française n° 87 (pp. 11-22). Larousse: Paris.
- Gross, M. (1975). *Méthodes en Syntaxe* (414 pages). Hermann: Paris.
- Gross, M. (1993). Les phrases figées en français. In *L’information grammaticale* (pp. 36-41). Paris.
- Leclère, C. (2002). Organization of the Lexicon-Grammar of French verbs. In *Linguisticae Investigationes* 25:1 (pp. 29-48). John Benjamins Publishing Company: Amsterdam/Philadelphia.
- Koeva S., Mihov S. (forthcoming 2005). Semantic Relations in INTEX. In *INTEX pour la Linguistique et le traitement automatique des langues* (2). Les Cahiers de la MSH Ledoux. Presses Universitaires de Franche-Comté: Besançon.
- Roche E. (1993). Une représentation par automate fini des textes et des propriétés transformationnelles des verbes. In *Linguisticae Investigationes* XVII :1 (pp. 189-222). John Benjamins Publishing Company: Amsterdam/Philadelphia.
- Silberztein, M. (1990). Le dictionnaire DELAC. In *Dictionnaires électroniques du français*, Langue française n° 87 (pp. 73-83). Larousse: Paris.
- Silberztein, M. (1993). Dictionnaires électroniques et analyse automatique de textes : le système INTEX. Masson: Paris.
- Silberztein, M. (1999). Traitement des expressions figées avec INTEX. In *Analyse lexicale et syntaxique : le système INTEX* (pp. 425-449). *Linguisticae Investigationes* XXII. John Benjamins Publishing Company: Amsterdam/Philadelphia.
- Silberztein M. (2004). NooJ: an oriented object approach. In *INTEX pour la Linguistique et le traitement automatique des langues*. Les Cahiers de la MSH Ledoux. Presses Universitaires de Franche-Comté: Besançon.
- Vietri Simona (2004). Lemmatization of Idioms in Italian. In *INTEX pour la Linguistique et le traitement automatique des langues*. Les Cahiers de la MSH Ledoux. Presses Universitaires de Franche-Comté: Besançon.